ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ
INFORMATION TECHNOLOGY, COMPUTER SCIENCE, AND MANAGEMENT



УДК 681.3.681.5 10.23947/1992-5980-2017-17-3-137-144

Сравнение эффективности работы точных и приближенных алгоритмов для решения задачи о покрытии множества*

И. С. Коновалов[1], С. С. Остапенко[2], В. Г. Кобак[3]**
[1,2,3] Донской государственный технический университет, г. Ростов-на-Дону, Российская Федерация

Efficiency comparison of exact and approximate algorithms for solving set covering problem***

I. S. Konovalov[1], S. S. Ostapenko[2], V. G. Kobak[3]**
[1,2,3] Don State Technical University, Rostov-on-Don, Russian Federation

Введение. Множество практических задач опирается на задачу покрытия множеств: построение расписаний, расположение пунктов обслуживания, построение электронных схем. Это определяет актуальность поиска способов повышения эффективности решения данной задачи.
Материалы и методы. Рассматриваются методы решения задачи о покрытии множества точным и приближенным алгоритмами. В качестве приближенного метода используется генетический алгоритм, в качестве точного — метод ветвей и границ.
Результаты исследования. Генетический алгоритм во всех своих модификациях по временным характеристикам показал предсказуемость и стабильность во всех сериях экспериментов. Метод ветвей и границ был применен к задаче покрытия множеств и показал точные результаты.
Обсуждение и заключения. Проведенные исследования показали, что для множеств небольших размеров целесообразно использовать метод ветвей и границ, который продемонстрировал быстрое время выполнения при гарантированно точном результате. Для множеств больших размеров рекомендуется использовать генетический алгоритм, который гарантирует результат с незначительной погрешностью, причем изменение времени его работы стабильно и предсказуемо.

Introduction. A quite general class of practical tasks is guided by the set covering problem: schedules building, layout of service stations, and creation of electronic circuits. It defines relevance of searching methods to improve the solution efficiency of this task.
Materials and Methods. Techniques of the set covering problem solution by exact and approximate algorithms are considered. The genetic algorithm is used as the approximate method, and the branch and bounds algorithm – as the exact method.
Research Results. The genetic algorithm in all its modifications on time response characteristics has shown predictability and stability in all series of experiments. The branch and bounds method was applied to the set covering task, and it has shown exact results.
Discussion and Conclusions. The conducted research shows that for small sets, it is expedient to use the branch and bounds method which has demonstrated fast runtime with an assured exact result. For large sets, it is recommended to use the genetic algorithm which guarantees receiving a result with a negligible error where the execution time shift is stable and predictable.

Ключевые слова: Задача покрытия множества, генетический алгоритм, модель Голдберга, алгоритм полного перебора, метод ветвей и границ, алгоритм Алексеева.

Keywords: set covering problem, genetic algorithm, Goldberg model, exhaustive algorithm, branch-and-bound method, Alekseev algorithm.

Образец для цитирования: Коновалов, И. С. Сравнение эффективности работы точных и приближенных алгоритмов для решения задачи о покрытии множества / И. С. Коновалов, С. С. Остапенко, В. Г. Кобак // Вестник Дон. гос. техн. ун-та. — 2017. — Т.17, № 3. — С. 137–144.

For citation: I.S. Konovalov, S.S. Ostapenko, V.G. Kobak. Efficiency comparison of exact and approximate algorithms for solving set covering problem. Vestnik of DSTU, 2017, vol. 17, no.3, pp. 137–144.

Введение. Наиболее важным классом задач в теории алгоритмов является класс *NP*-полных задач. На сегодняшний день нахождение оптимальных решений в этой области не перестает быть актуальным. В частности, большое количество прикладных задач можно свести к задаче о покрытии множества. Примерами могут являться многие задачи дискретной оптимизации: задачи стандартизации, упаковки и разбиения множества, задача о наибольшей клике, построение расписаний. Известна также и обратная сводимость задачи о покрытии к этим задачам.

* Работа выполнена в рамках инициативной НИР.
** E-mail: e-mail:xigorx92@mail.ru, cj-x@yandex.ru, valera33305@mail.ru
*** The research is done within the frame of the independent R&D.

На практике задачи о покрытии возникают при размещении пунктов обслуживания, в системах информационного поиска, при назначении экипажей на транспорте, проектировании интегральных схем и конвейерных линий и т. д.

Постановка задачи. Дано множество U из n элементов и набор подмножеств U, $S=\{S_1,\ldots, S_k\}$. Каждому подмножеству S_i сопоставлена некоторая неотрицательная стоимость c: $S\to Q^+$. $S'\subseteq S$ является покрытием, если любой элемент из U принадлежит хотя бы одному элементу из S' [1–3].

Задача о покрытии множествами заключается в нахождении набора подмножеств, покрывающего все множество U и имеющего минимально возможный вес (в случае взвешенной задачи) или минимально возможное число подмножеств (в случае невзвешенной задачи).

Можно представить задачу в матричном виде [4, 5]. Пусть $A=(a_{ij})$ — произвольная матрица размера mxn с элементами $a_{ij}\in\{0,1\}$ без нулевых строк и столбцов. Будем говорить, что в A строка i покрывается столбцом j, если $a_{ij}=1$. Подмножество столбцов называется покрытием, если в совокупности они покрывают все строки матрицы A. Пусть каждому столбцу поставлено в соответствие положительное число c_j, называемое весом столбца. Требуется найти покрытие минимального суммарного веса. Вводя переменные x_j, равные единице, если столбец j входит в искомое покрытие, и равные нулю в противном случае, приходим к следующей формулировке задачи о покрытии:

$$\sum_{j=1}^{n} c_j x_j \to \min, \tag{1}$$

при ограничениях

$$\sum_{j=1}^{n} a_{ij} x_j \ge 1, i=1,\ldots m, x_j \in \{0,1\}, j=1,\ldots,n \ . \tag{2}$$

Целью данного исследования является выявление целесообразности использования некоторых алгоритмов для решения задачи о покрытии множества на основе сравнения эффективности их работы по критериям быстродействия и точности. Для сравнения были выбраны: генетический алгоритм, метод ветвей и границ, алгоритм полного перебора.

Генетический алгоритм. Генетические алгоритмы основаны на идеях естественного отбора и наследования и относятся к вероятностным эвристическим методам [6–9].

В настоящем исследовании используется модифицированная модель Голдберга: оптимизационной функцией является минимизация веса покрытия, а условием останова — неизменность лучшего решения в течение заданного числа поколений. Особенностью исследуемого алгоритма является то, что в кроссовере участвуют все особи текущего поколения. Кроме того, для каждого потомка применяется оператор мутации.

Рассмотрим механизм кодирования особи. Каждый индивид k представлен хромосомой, являющейся n-мерным вектором x_k, у которого j-й элемент x_{kj} принимает значение 1, если подмножество S_j входит в покрытие, а в ином случае принимает значение 0. С таким представлением степень приспособленности f_k индивида x_k может быть рассчитана следующим образом:

$$f_k = \sum_{j=1}^{n} c_j x_j^k, \tag{3}$$

где c_j — стоимость подмножества S_j.

Таким образом, оптимизационная функция выглядит как $f_k\to\min$.

Для выбора родительских особей используется случайный отбор. Также применяется стратегия элитного отбора для улучшения точности результата без временных затрат: элитная особь формируется в первом поколении с помощью жадного алгоритма, описанного в статье [1].

В алгоритме используется тип мутации, основанный на изменении случайного гена на противоположное значение. Оператор скрещивания точечный. Выбираются пары хромосом из родительской популяции. Далее для каждой пары отобранных таким образом родителей разыгрывается позиция гена (локус) в хромосоме, определяющая так называемую точку скрещивания — l_k. В результате скрещивания пары родительских хромосом получается следующая пара потомков: P_1 — потомок, хромосома которого на позициях от 1 до l_k состоит из генов первого родителя, а на позициях от l_k+1 до L — из генов второго родителя; P_2 — потомок, хромосома которого на позициях от 1 до l_k состоит из генов второго родителя, а на позициях от l_k+1 до L — из генов первого родителя. Начальное поколение формируется из особей, соответствующих найденным случайным образом покрытий.

Особенность работы генетического алгоритма состоит в том, что при скрещивании и мутации могут появиться особи, соответствующие покрытия которых не существуют, то есть являющиеся недопустимыми решениями. От этих решений необходимо избавляться. Алгоритм проверяет, существует ли покрытие, и если нет, то пытается, в случае скрещивания, выбрать другую вторую родительскую особь, а в случае мутации — выбрать другой

ген для его инвертирования. Если и это не «исправит» особь, родитель выберется заново случайным образом. Потомок заменяет случайно выбранную особь, если его приспособленность выше.

Алгоритм полного перебора (брутфорс). Полный перебор — точный метод решения оптимизационных задач, относящийся к классу методов поиска решения исчерпыванием всевозможных вариантов. Сложность полного перебора зависит от количества всех возможных решений задачи.

Полным перебором можно решить любую задачу из класса *NP*-полных задач. В зависимости от количества всех возможных решений задачи и времени вычисления целевой функции от каждого решения полный перебор может потребовать экспоненциального времени работы.

Идея алгоритма полного перебора для решения задачи покрытия заключается в следующем:

1) поиск всех возможных сочетаний подмножеств исходного множества и сравнение их целевых функций;
2) определение покрытий среди найденных сочетаний;
3) нахождение покрытия минимального веса.

Из теории множеств известно, что число всех подмножеств множества из n элементов равно 2^n. Другими словами,

$$\sum_{k=0}^{n} C_n^k = 2^n. \quad (4)$$

Метод ветвей и границ. Данный метод является модификацией алгоритма полного перебора, что гарантирует точность результата его работы. Суть метода заключается в построении дерева полного перебора и отсечении бесперспективных ветвей решения по мере его обхода, что существенно уменьшает время его работы [10]. Условия задачи описываются формулами (1) и (2). Дерево перебора в этом случае строится так, что на каждом уровне k для каждого из узлов k–1 уровня добавляются в качестве дочерних узлов все возможные варианты $a_{kj} \cdot x_j$. Для поставленной задачи на каждом уровне добавляется количество узлов, равное количеству $a_{kj} \neq 0$. Алгоритм полного перебора предполагает полный обход такого дерева. Алгоритм после обработки каждого узла k-ого уровня приступает к обработке дочерних узлов прежде, чем переходит к следующему узлу k-ого уровня. Метод ветвей и границ сокращает время поиска оптимального решения за счет того, что при вхождении в каждый узел выполняется верхняя и/или нижняя оценка возможного решения, к которому приведет обход поддерева, корнем которого является текущий узел. В соответствии с полученной оценкой делается вывод — если лучшее из возможных решений хуже текущего, то данное поддерево (ветвь) отсекается и обход продолжается со следующего узла того же уровня, на котором было произведено отсечение.

Таким образом, ключевым аспектом работы алгоритма по методу ветвей и границ является эффективная оценка поддерева, обязательным требованием к которой также является недопустимость потери точности. Поскольку алгоритм оценки поддерева опирается на текущее решение, инициализация начального решения оставляется на усмотрение разработчика. Возможны четыре варианта инициализации:

1) заполнение решения максимальными значениями;
2) выбор случайного покрытия;
3) последовательная инициализация лучшими значениями (алгоритм Алексеева);
4) нахождение решения приближенным алгоритмом.

Поскольку многообразие и эффективность работы приближенных алгоритмов являются отдельным предметом исследования, для данного исследования их использование в качестве начального значения недопустимо. В связи с этим был выбран вариант инициализации по алгоритму Алексеева: последовательно, на каждом шаге в решение добавляется такая переменная x_j, которая дает на данном шаге лучшее решение. Если таких переменных несколько, выбирается первая найденная.

Для задачи покрытия множеств был разработан следующий оценочный метод. Пусть N — текущее решение задачи, то есть минимальное найденное количество подмножеств S_i, покрывающих исходное множество U. В таком случае, минимальное улучшение равняется N–1. Текущим состоянием M назовем число подмножеств S_i, покрывающих сформированный на данном этапе набор $x_1 x_2 \dots x_j$. При $M=N-1$ выполняется проверка: если среди непокрытых переменных x_j есть хотя бы одна, которая не покрывается подмножествами S_i, формирующими состояние M, то минимальное улучшение невозможно, следовательно, данная ветвь считается бесперспективной и отбрасывается. В противном случае текущее состояние M становится текущим решением задачи.

Входные данные и исходные параметры алгоритмов. В качестве входных данных для всех алгоритмов взята матрица A, представленная формулой (2).

Для сравнения эффективности работы были выбраны генетический алгоритм и метод ветвей и границ.

В качестве исходных параметров генетического алгоритма установлены: вероятность скрещивания — 100%, вероятность мутации — 100%. Для исследования выбрано 3 варианта с различным количеством особей — 100, 200 и

300. Для улучшения точности результата используется принцип элитного отбора. Для метода ветвей и границ начальное решение формируется алгоритмом Алексеева.

Подтверждение точности метода ветвей и границ. С целью подтверждения точности алгоритма по методу ветвей и границ, а также доказательства его эффективности по временным показателям, были проведены серии экспериментов по сравнению его работы с работой алгоритма полного перебора на матрицах различного размера.

В представленных ниже таблицах приведены усредненное значение результата и среднее время работы в миллисекундах, алгоритм полного перебора обозначен «АПП», метод ветвей и границ — «ВиГ».

Таблица 1

Table 1

Результаты экспериментов при размере матрицы А — 10х10; количестве экспериментов — 500

Results of experiments with A matrix size — 10x10; number of experiments — 500

	АПП	ВиГ
Результат	2,288	2,288
Время, мс	0,8343	0,0181

Таблица 2

Table 2

Результаты экспериментов при размере матрицы А — 15х15; количестве экспериментов — 500

Results of experiments with A matrix size — 15x15; number of experiments — 500

	АПП	ВиГ
Результат	2,398	2,398
Время, мс	56,6256	0,0617

Таблица 3

Table 3

Результаты экспериментов при размере матрицы А — 20х20; количестве экспериментов — 500

Results of experiments with A matrix size — 20x20; number of experiments — 500

	АПП	ВиГ
Результат	2,612	2,612
Время, мс	3133,0340	0,1183

Таблица 4

Table 4

Результаты экспериментов при размере матрицы А — 25х25; количестве экспериментов — 500

Results of experiments with A matrix size — 25x25; number of experiments — 500

	АПП	ВиГ
Результат	2,614	2,614
Время, мс	155340,0994	0,2492

Четыре серии экспериментов (суммарно 1600) не выявили ни одного расхождения результатов работы алгоритмов полного перебора и метода ветвей и границ. Таким образом, точность метода ветвей и границ доказана не только теоретически, но и экспериментально.

По временным показателям метод ветвей и границ сработал быстрее во всех 4 сериях экспериментов: для матрицы размером 10х10 — в 46 раз, 15х15 — в 918 раз, 20х20 — в 26 480 раз и для 25х25 — в 623 451 раз.

Данные результаты позволяют сделать однозначный вывод о преимуществе использования метода ветвей и границ перед алгоритмом полного перебора и в дальнейшем в исследовании в качестве точного алгоритма будет использоваться только метод ветвей и границ.

Сравнение эффективности работы точных и приближенных алгоритмов. В данном исследовании для сравнения было выбрано 4 алгоритма: 3 модификации генетического алгоритма с использованием элиты, с количествами особей 100 (ГА100), 200 (ГА200) и 300 (ГА300), а также алгоритм по методу ветвей и границ (ВиГ).

В представленных ниже таблицах приведены усредненное значение результата (для вычисления среднего отклонения) и среднее время работы в миллисекундах.

Таблица 5
Table 5

Результаты экспериментов при размере матрицы А — 20x20; количестве экспериментов — 500
Results of experiments with A matrix size — 20x20; number of experiments — 500

	ГА100	ГА200	ГА300	ВиГ
Результат	2,688	2,684	2,684	2,684
Время, мс	426,154	1033,16	1435,32	0,18

Таблица 6
Table 6

Результаты экспериментов при размере матрицы А — 50x50; количестве экспериментов — 500
Results of experiments with A matrix size — 50x50; number of experiments — 500

	ГА100	ГА200	ГА300	ВиГ
Результат	3,112	3,024	3	3
Время, мс	1406,05	3443,07	5266,25	8,21

Таблица 7
Table 7

Результаты экспериментов при размере матрицы А — 75x75; количестве экспериментов — 500
Results of experiments with A matrix size — 75x75; number of experiments — 500

	ГА100	ГА200	ГА300	ВиГ
Результат	3,846	3,684	3,572	3,142
Время, мс	1724,13	3553,52	5437,28	320,64

Таблица 8
Table 8

Результаты экспериментов при размере матрицы А — 85x85; количестве экспериментов — 200
Results of experiments with A matrix size — 85x85; number of experiments — 200

	ГА100	ГА200	ГА300	ВиГ
Результат	4,01	3,925	3,85	3,445
Время, мс	2030,56	4159,29	6354,38	8851,48

Таблица 9
Table 9

Результаты экспериментов при размере матрицы А — 85x85; количестве экспериментов — 400
Results of experiments with A matrix size — 85x85; number of experiments — 400

	ГА100	ГА200	ГА300	ВиГ
Результат	3,904	3,8275	3,8225	3,395
Время, мс	2202,56	4312,41	6394,33	2892,97

Таблица 10
Table 10

Результаты экспериментов при размере матрицы А — 90x90; количестве экспериментов — 300
Results of experiments with A matrix size — 90x90; number of experiments — 300

	ГА100	ГА200	ГА300	ВиГ
Результат	3,96	3,93	3,91	3,56
Время, мс	2080,22	4735,22	7088,39	1926,04

Таблица 11
Table 11

Результаты экспериментов при размере матрицы А — 100x100; количестве экспериментов — 100
Results of experiments with A matrix size — 100x100; number of experiments — 100

	ГА100	ГА200	ГА300	ВиГ
Результат	3,98	3,99	3,95	3,79
Время, мс	4448,63	9450,40	14296,89	3655,89

Таблица 12
Table 12

Результаты экспериментов при размере матрицы А — 100x100; количестве экспериментов — 200
Results of experiments with A matrix size — 100x100; number of experiments — 200

	ГА100	ГА200	ГА300	ВиГ
Результат	4,015	3,995	3,924	3,85
Время, мс	4374,55	9291,33	15378,67	38932,69

Анализ полученных результатов. В сериях экспериментов 1, 2 и 3 точный алгоритм по методу ветвей и границ показал убедительное преимущество в скорости работы, превзойдя самую быструю модификацию генетического алгоритма (ГА100) в 2367,5 раз в первом случае, в 171,26 во втором и в 52,82 в третьем.

Однако, начиная с четвертой серии экспериментов, время работы алгоритма по методу ветвей и границ сильно варьируется и показывает весь спектр результатов от лучшего к худшему. Объясняется это тем, что ход работы алгоритма сильно зависит от входной матрицы. При большой плотности единиц разветвленность дерева больше, а также больше ветвей, в которых оценка возможна лишь ближе к листьям. То есть, несмотря на то, что время работы метода ветвей и границ будет всегда меньше, чем время работы алгоритма полного перебора T, в худшем случае время данного метода может стремиться к $\frac{T}{n}$, где n — некоторая константа.

Генетический алгоритм во всех модификациях по временным характеристикам показал предсказуемость и стабильность во всех сериях экспериментов. Среднее отклонение от точного результата для ГА100 составило 9,77%, для ГА200 — 8,13%, для ГА300 — 6,68%. Вышеизложенные результаты позволяют сделать вывод о том, что с ростом количества особей улучшается точность генетического алгоритма.

Выводы. Проведенные исследования показали, что для множеств небольших размеров (до 75 элементов включительно) целесообразно использовать метод ветвей и границ, который продемонстрировал высокое быстродействие при гарантии точного результата. Для множеств больших размеров рекомендуется использовать генетический алгоритм, который гарантирует получение результата с незначительной погрешностью, но за определенный фиксированный промежуток времени. Метод же ветвей и границ, по результатам проведенных экспериментов, с увеличением размера входных данных может вести себя непредсказуемо.

Библиографический список

1. Коновалов, И. С. Сравнительный анализ работы жадного алгоритма Хватала и модифицированной модели Голдберга при решении взвешенной задачи нахождения минимального покрытия множеств / И. С. Коновалов, В. А. Фатхи, В. Г. Кобак // Труды СКФ МТУСИ. — Ростов-на-Дону: ПЦ «Университет» СКФ МТУСИ, 2015 — С. 366–371.

2. Коновалов, И. С. Стратегия элитизма модифицированной модели Голдберга генетического алгоритма при решении задачи покрытия множеств / И. С. Коновалов, В. А. Фатхи, В. Г. Кобак // Вестник компьютерных и информационных технологий. — 2016. — №4. — С. 50–56.

3. Коновалов, И. С. Применение генетического алгоритма для решения задачи покрытия множеств / И. С. Коновалов, В. А. Фатхи, В. Г. Кобак // Вестник Дон. гос. техн. ун-та. — 2016. — № 3. — С. 125–132.

4. Еремеев, А. В. Генетический алгоритм для задачи о покрытии / А. В. Еремеев // Дискретный анализ и исследование операций. — 2000. — Т. 7, № 1. — С. 47–60.

5. Еремеев, А. В. Задача о покрытии множества: сложность, алгоритмы, экспериментальные исследования / А. В. Еремеев, Л. А. Заозерская, А. А. Колоколов // Дискретный анализ и исследование операций. — 2000. — Т. 7, № 2. — С. 22–47.

6. Holland, J. H. Adaptation in Natural and Artificial Systems. — The University of Michigan Press, 1975. — P. 245.

7. Goldberg, D. E. Genetic algorithms in search, optimization and machine learning. Reading, MA: Addison-Wesley, 1989. — P. 432.

8. Батищев, Д. И. Генетические алгоритмы решения экстремальных задач / Д. И. Батищев. — Воронеж : изд-во Воронеж. гос. техн. ун-та, 1995. — 121 с.

9. Гладков, Л. А. Генетические алгоритмы / Л. А. Гладков, В. В. Курейчик, В. М. Курейчик. — Москва : ФИЗМАТЛИТ, 2010. — 368 с.

10. Алексеев, О. Г. Некоторые алгоритмы решения задачи о покрытии и их экспериментальное исследование на ЭВМ / О. Г. Алексеев, В. Ф. Григорьев // Журнал вычислительной математики и математической физики. — 1984. — Т. 24, №10. — С. 1565–1570.

References

1. Konovalov, I.S., Fatkhi, V.A., Kobak, V.G. Sravnitel'nyy analiz raboty zhadnogo algoritma Khvatala i modifitsirovannoy modeli Goldberga pri reshenii vzveshennoy zadachi nakhozhdeniya minimal'nogo pokrytiya mnozhestv. [Comparative analysis of work of Chvatal greedy algorithm and modified Goldberg model under solving the problem on determining minimal set covering.] Trudy SKF MTUSI. Rostov-on-Don: PTs «Universitet» SKF MTUSI, 2015, pp. 366–371 (in Russian).

2. Konovalov, I.S., Fatkhi, V.A., Kobak, V.G. Strategiya elitizma modifitsirovannoy modeli Goldberga geneticheskogo algoritma pri reshenii zadachi pokrytiya mnozhestv. [Elitism strategy of modified Goldberg model of genetic algorithm in solving the set covering problem.] Herald of Computer and Information Technologies, 2016, no. 4, pp. 50–56 (in Russian).

3. Konovalov, I.S., Fatkhi, V.A., Kobak, V.G. Primenenie geneticheskogo algoritma dlya resheniya zadachi pokrytiya mnozhestv. [Application of genetic algorithm for the set-covering problem solution.] Vestnik of DSTU, 2016, no. 3pp. 125–132 (in Russian).

4. Eremeev, A.V. Geneticheskiy algoritm dlya zadachi o pokrytii. [A genetic algorithm for the covering problem.] Discrete Analysis and Operations Research, 2000, ser. 2, vol. 7, no. 1, pp. 47–60 (in Russian).

5. Eremeev, A.V., Zaozerskaya, L.A., Kolokolov, A.A. Zadacha o pokrytii mnozhestva: slozhnost', algoritmy, eksperimental'nye issledovaniya. [The set covering problem: complexity, algorithms, and experimental investigations.] Discrete Analysis and Operations Research, 2000, ser.2, vol. 7, no. 2, pp. 22–47 (in Russian).

6. Holland, J.H. Adaptation in Natural and Artificial Systems. The University of Michigan Press, 1975, p. 245.

7. Goldberg, D.E. Genetic algorithms in search, optimization and machine learning. Reading, MA: Addison-Wesley, 1989, p. 432.

8. Batishchev, D.I. Geneticheskie algoritmy resheniya ekstremal'nykh zadach. [Genetic algorithms for solving extremal problems.] Voronezh: izd-vo Voronezh. gos. tekhn. un-ta, 1995, 121 p. (in Russian).

9. Gladkov, L.A., Kureychik, V.V., Kureychik, V.M. Geneticheskie algoritmy. [Gtnetic algorithms.] Moscow: FIZMATLIT, 2010, 368 p. (in Russian).

10. Alekseev, O.G., Grigoryev, V.F. Nekotorye algoritmy resheniya zadachi o pokrytii i ikh eksperimental'noe issledovanie na EVM. [Some algorithms for solving the covering problem and their experimental computed study.] Computational Mathematics and Mathematical Physics, 1984, vol. 24, no. 10, pp. 1565–1570 (in Russian).

Поступила в редакцию 22.05.2017
Сдана в редакцию 23.05.2017
Запланирована в номер 17.07.2017

Received 22.05.2017
Submitted 23.05.2017
Scheduled in the issue 17.07.2017

Об авторах:

Коновалов Игорь Сергеевич,
аспирант кафедры «Программное обеспечение вычислительной техники и автоматизированных систем» Донского государственного технического университета (РФ, 344000, Ростов-на-Дону, пл. Гагарина, 1),
ORCID: http:// orcid.org/0000-0001-6296-3660
xigorx92@mail.ru

Остапенко Сергей Сергеевич,
аспирант кафедры «Программное обеспечение вычислительной техники и автоматизированных систем» Донского государственного технического университета (РФ, 344000, Ростов-на-Дону, пл. Гагарина, 1),
ORCID: http:// orcid.org/0000-0002-8711-5059
cj-x@yandex.ru

Кобак Валерий Григорьевич,
профессор кафедры «Программное обеспечение вычислительной техники и автоматизированных систем» Донского государственного технического университета (РФ, 344000, Ростов-на-Дону, пл. Гагарина, 1), доктор технических наук профессор,
ORCID: http:// orcid.org/0000-0002-1001-0574
valera33305@mail.ru

Authors:

Konovalov, Igor S.,
postgraduate student of the Computer and Automated Systems Software Department, Don State Technical University (RF, 344000, Rostov-on-Don, Gagarin Square, 1),
ORCID: http:// orcid.org/0000-0001-6296-3660
xigorx92@mail.ru

Ostapenko, Sergey S.,
postgraduate student of the Computer and Automated Systems Software Department, Don State Technical University (RF, 344000, Rostov-on-Don, Gagarin Square, 1),
ORCID: http:// orcid.org/0000-0002-8711-5059
cj-x@yandex.ru

Kobak, Valery G.,
associate professor of the Computer and Automated Systems Software Department, Don State Technical University (RF, 344000, Rostov-on-Don, Gagarin Square, 1), Dr.Sci. (Eng.), professor,
ORCID: http:// orcid.org/0000-0002-1001-0574
valera33305@mail.ru